

General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.
- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.
- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.
- This document is paginated as submitted by the original source.
- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

Lockheed Electronics Company, Inc.

A SUBSIDIARY OF
LOCKHEED CORPORATION

1830 NASA Road 1, Houston, Texas 77058
Tel. 713-333-5411

JSC-14790

APR 5 1979

CR160161

Ref: 642-6963
NAS 9-15800
Job Order 73-743-20

TECHNICAL MEMORANDUM

USER'S GUIDE FOR THE CATE-LIEBIG LAW OF THE MINIMUM

By

D. E. Phinney and R. B. Cate

(NASA-CR-160161) USER'S GUIDE FOR THE
CATE-LIEBIG LAW OF THE MINIMUM (Lockheed
Electronics Co.) 53 p HC A04/MF A01

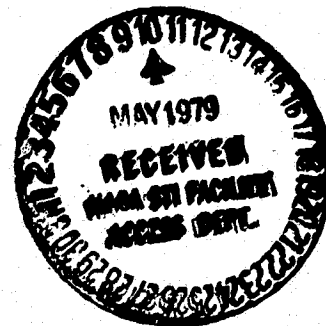
N79-21801

CSCL 09B

63/61

Unclas
24525

Approved By: B. L. Carroll
B. L. Carroll, Manager
EO Development and Evaluation
Department



February 1979

LEC-12377

CONTENTS

Section	Page
1. GENERAL DESCRIPTION OF THE COMPUTER PROGRAM	1-1
2. USER'S GUIDE	2-1
2.1 <u>PROGRAM MODES</u>	2-1
2.2 <u>PROGRAM OUTPUT</u>	2-1
2.3 <u>LIMITATIONS PER PROBLEM</u>	2-1
2.4 <u>ORDER OF CARDS IN JOB DECK</u>	2-2
2.5 <u>CARD PREPARATION</u>	2-2
Appendix	
A. PROGRAM LISTING.	A-3
B. SAMPLE INPUT DATA.	B-3
C. SAMPLE OUTPUT DATA	C-3

1. GENERAL DESCRIPTION OF THE COMPUTER PROGRAM

This computer program is used to implement the Cate-Liebig algorithm for continuous data. The Cate-Liebig approach is intended to provide an alternative to stepwise multiple linear regression for the development of predictive models. In the program, cases are allocated to individual variables in an iterative manner based on the zero means of optimum binary (0-1) dummy variables created by using the Cate-Nelson critical level algorithm. The means of the allocated cases, along with a calculated overall maximum applied at the final critical level, are utilized to define a linear response surface with an independent equation for each variable.

The predicted value for any case is the minimum of all the values predicted for that case by the various individual equations. Variables are deleted from the model if less than a user-specified number of cases is allocated to that variable. Provision is made for testing the model on an independent set of data.

This documentation and the program structure are modeled after the widely known biomedical (BMD) computer programs developed by the University of California. It is intended to be readily usable for persons familiar with the BMD program package; however, this documentation may be used without additional reference to any other publication.

2. USER'S GUIDE

2.1 PROGRAM MODES

The program may be run in two primary modes. In the first mode, the program fits a set of dependent data using the Cate-Liebig algorithm and optionally provides a test of the resulting model on an independent data set. In the second mode, the program runs a user-specified model on an independent data set.

2.2 PROGRAM OUTPUT

Output from this program includes the following:

1. Summary of allocations.
2. Model coefficients.
3. For fit and test data (if any):
 - a. Limiting variable for each case.
 - b. Estimate, actual value, and error of dependent variable.
 - c. Simple error statistics, including mean error, mean square error, root mean square error, and coefficient of determination.
 - d. Plot of estimated versus actual value of the dependent variable.
4. Optional output, such as
 - a. Plot of raw data for each variable.
 - b. Summary of consistency check.
 - c. Summary of critical-level splits after each iteration.
 - d. Plot of cases allocated to each variable.

2.3 LIMITATIONS PER PROBLEM

Limitations per problem are as follows:

1. Number of original variables (NOV); $1 \leq \text{NOV} \leq 80$

2. Total number of variables after transgeneration (IP); $1 \leq IP \leq 80$
3. Number of variable format cards (NVFC); $1 \leq NVFC \leq 10$
4. Number of cases in fit (N); $2 \leq N \leq 2000$
5. Number of test cases (NTC); $1 \leq NTC \leq 2000$
6. Number of transgeneration cards (NTGC); $0 \leq NTGC \leq 99$

This program allows transgeneration of variables.

2.4 ORDER OF CARDS IN JOB DECK

Cards in the job deck should appear in the following order (NOTE: items b through g may be repeated as often as desired):

- a. Program deck (a listing of the complete Fortran program is given in appendix A. All of the following cards are data read by that program.)
- b. Problem card
- c. Transgeneration cards (optional)
- d. "Labels" cards (optional)
- e. F-type variable format card(s)
- f. Data deck (place data input deck here, including test cases)
- g. Control-delete cards (optional). Alternatively when run in the model test mode only, the model definition cards are input at this point.
- h. Finish card

2.5 CARD PREPARATION

- a. System cards (to be determined)
- b. Problem card:

<u>Column</u>	<u>Problem</u>
1-6	Problem (mandatory)
10-15	Alphanumeric problem name
17-20	Number of cases

<u>Column</u>	<u>Description</u>
24-25	Number of original variables
29-30	Number of transgeneration cards
34-35	Number of variables added by transgeneration
39-40	Index of dependent variable
44-45	Number of control-delete cards
49-50	Number of labeled variables
52-53	Minimum R^2 on split for allocation
54-55	Do not plot raw data if NPS = 1
57-58	Consistency check will not be used if = 1
59-60	Print summary of splits for each split if = 1
64-65	Do not print and plot each allocated variable if = 1
66-70	Number of test cases
71-72	Number of variable format cards
73-75	Smallest number of observations allowed for a variable to be allocated
76-78	Index of weighting vector

NOTE: Any input variable may be used. All cases are assumed to be weighted equally if index is not given a value.

If the number of cases = 0, the program is run in the model test mode and the model definition cards are required.

c. Transgeneration card:

<u>Column</u>	<u>Description</u>
1-6	TRNGEN
7-9	Variable index K
10-11	Code from transgeneration list
12-14	Variable index I
15-20	Variable index J or constant C

The term *transgeneration* is used to include transformations of input variables and creation of new variables prior to computation. Notations to be used in the transgeneration list include the following.

i, j, k = variable indices (need not be different)

c = a constant

a_1, a_2, a_3, \dots = constants

n = the number of cases, or sample size

<u>Code</u>	<u>Transgeneration</u>	<u>Restriction</u>
01	$\sqrt{X_i} \rightarrow X_k$	$X_i \geq 0$
02	$\sqrt{X_i} + \sqrt{X_i + 1} \rightarrow X_k$	$X_i \geq 0$
03	$\log_{10} X_i \rightarrow X_k$	$X_i \geq 0$
04	$e^{X_i} \rightarrow X_k$	
05	$\arcsin \sqrt{X_i} \rightarrow X_k$	$0 \leq X_i \leq 1$
06	$\arcsin \sqrt{X_i/(n+1)}$ $+ \arcsin \sqrt{(X_i+1)/(n+1)} \rightarrow X_k$	$0 \leq (X_i/n) \leq 1$
07	$1/X_i \rightarrow X_k$	$X_i \neq 0$
08	$X_i + c \rightarrow X_k$	None
09	$X_i c \rightarrow X_k$	None
10	$X_i^c \rightarrow X_k$	$X_i \geq 0$
11	$X_i + X_j \rightarrow X_k$	None
12	$X_i - X_j \rightarrow X_k$	None
13	$X_i X_j \rightarrow X_k$	None
14	$X_i/X_j \rightarrow X_k$	$X_j \neq 0$
15	If $X_i > c, 1 \rightarrow X_k$; otherwise $0 \rightarrow X_k$	None

<u>Code</u>	<u>Transgeneration</u>	<u>Restriction</u>
16	If $X_i \geq X_j$, $1 \rightarrow X_k$; otherwise $0 \rightarrow X_k$	None
17	$\log_e X_i \rightarrow X_k$	$X_i > 0$
18	If $X_i \leq c$, $1 \rightarrow X_k$; otherwise $0 \rightarrow X_k$	None
20	$\sin X_i \rightarrow X_k$	None
21	$\cos X_i \rightarrow X_k$	None
22	$\text{arc tan } X_i \rightarrow X_k$	None
23	$X_i^{X_j} \rightarrow X_k$	$X_i > 0$
24	$c^{X_i} \rightarrow X_k$	$c > 0$

When a violation of a restriction in the right-hand column occurs during transgeneration, the program will print a diagnostic message.

d. Label cards:

<u>Column</u>	<u>Description</u>
1-6	Labels (mandatory)
7-10	Number of the variable (or category or index) to be named (this number must be right-justified)
11-16	Corresponding alphanumeric name
17-20	Number of another variable
21-26	Corresponding alphanumeric name
⋮	
67-70	Number of another variable
71-76	Corresponding alphanumeric name of that variable (up to 7 variables per card)

Label cards allow the user to substitute alphanumeric names for the usual numeric indices that appear on the printed output. There may be from one to seven pairs of variable numbers and labels on each "labels" card. If desired, only one pair may be specified on each card. However, the total number of labels appearing on all the "labels" cards must be equal the number of labels specified on the problem card. It is not necessary to label all the variables. Those labeled may be listed in any order.

- e. F-type variable format cards: The format is keypunched beginning with a left parenthesis, followed by a sequence of specifications, and closed by a right parenthesis. The first two specifications below are followed by a comma, except when they precede a slash or a right-hand parenthesis. Blank columns within the format are ignored, and columns 73 to 80 are not used. (A complete description of formats can be found in any Fortran programming manuals.) The specifications for F-type variable format cards are as follows:

- "nFw.d" is the floating point (decimal) indicator; n is the number of fields of width w (includes a sign and decimal point if punched); and d is the number of digits to the right of the decimal point if the decimal is not punched ($0 \leq d \leq w$). If the decimal is punched, d is ignored. If n is not specified, it is assumed to be 1.
- "mX" (alphabetic X), in which X is the skip indicator and m is the number of columns to be skipped.
- "/" (slash) indicates, "go to the next data card." Depending on its location in the format statement, the "/" will either direct the program to go immediately to the next card (ignoring any further information on the current card) or skip one card altogether. For example, if a format begins with " /," the program will automatically skip the first card, read the second, skip the third card, etc. If a format ends with " /," the program will automatically read the first card, skip the second, read the third card, etc.
- "//" indicates, "go to the card after next." Two slashes "//" will direct the program to skip two cards. Any number of slashes may be used.

f. Control-delete cards

<u>Column</u>	<u>Description</u>
1-6	CØNDEL
7	Control value for first variable
8	Control value for second variable
⋮	
72	Control value for 66th variable

If there are more than 66 variables, continue on another card of the same form until all variables have been specified. The variable numbers above refer to variables after transgeneration.

NOTE: The following control values are permissible:

- 0 — Linear form will be assumed.
- 1 — Delete variable (or dependent variable).
- 2 — A 0-1 form will be assumed.

If no control-delete cards are included or if a field is left blank on the control-delete cards included in the deck, the value 0 will be assigned if the variable is not the dependent variable and the value 1 will be assigned if it is the dependent variable.

- g. Model definition cards: If the number of cases on the problem card is set equal to zero, the control delete card(s) is (are) replaced by Card 1:

<u>Column</u>	<u>Description</u>
1-3	Number of variables in the model (integer)
4-14	Maximum Y response in model (floating point)

This card is followed by one card for each variable in the model in the following format.

<u>Column</u>	<u>Description</u>
1-3	Variable number after transgeneration (integer)

<u>Column</u>	<u>Description</u>
4-14	Intercept (linear model) or lower mean 0-1 model (floating point)
15-25	Slope (floating point)
26-36	Critical level (floating point)
37-40	Type of response function for this variable ($\pm 1, 0$)

If the code for the type of response is set equal to zero, a linear model of the $A + BX_i$ will be assumed. If the slope, B, is positive,

$$Y_i = A + BX_i \quad X_i \leq \text{critical level}$$

$$Y_i = \text{maximum} \quad \text{otherwise}$$

If the slope is negative,

$$Y_i = A + BX_i \quad X_i > \text{critical level}$$

$$Y_i = \text{maximum} \quad \text{otherwise}$$

If the code for the type of response is set equal to ± 1 , a 0-1 mean model will be assumed. If the code equal +1,

$$Y_i = \text{lower mean} \quad X_i \leq \text{critical level}$$

$$Y_i = \text{maximum} \quad \text{otherwise}$$

If the code equals -1,

$$Y_i = \text{lower mean} \quad X_i > \text{critical level}$$

$$Y_i = \text{maximum} \quad \text{otherwise}$$

Note that the program output tabulates all of these values in the model summary. When a 0-1 form has been assumed in the model fit, the appropriate lower mean value will be listed as the intercept.

The form of the response functions may be mixed on a variable-by-variable basis.

g. Finish card

Column

Description

1-6

Finish

This card will notify the program that the entire job is finished.

APPENDIX A
PROGRAM LISTING

APPENDIX A
PROGRAM LISTING

A listing of the complete Fortran program is given on the following pages.

CAT000010
CAT000020
CAT000030
CAT000040
CAT000050
CAT000060
CAT000070
CAT000080
CAT000090
CAT000100
CAT000110
CAT000120
CAT000130
CAT000140
CAT000150
CAT000160
CAT000170
CAT000180
CAT000190
CAT000200
CAT000210
CAT000220
CAT000230
CAT000240
CAT000250
CAT000260
CAT000270
CAT000280
CAT000290
CAT000300
CAT000310
CAT000320
CAT000330
CAT000340
CAT000350
CAT000360
CAT000370
CAT000380
CAT000390
CAT000400
CAT000410
CAT000420
CAT000430
CAT000440
CAT000450
CAT000460
CAT000470
CAT000480
CAT000490
CAT000500
CAT000510
CAT000520
CAT000530
CAT000540
CAT000550
CAT000560
CAT000570
CAT000580
CAT000590
CAT000600
CAT000610
CAT000620
CAT000630
CAT000640
CAT000650
CAT000660
CAT000670
CAT000680
CAT000690
CAT000700
CAT000710
CAT000720
CAT000730
CAT000740
CAT000750
CAT000760
CAT000770
CAT000780

**ORIGINAL PAGE IS
OF POOR QUALITY**

FILE. . . CLO516 FORTRAN P1

```

7008 IF(INPA.NE.1) WRITE(6,7008) CAT00790
      FORMAT(2X,30HPRINT AND PLOT EACH ALLOCATION ) CAT00800
      IF(NS.FO.1) WRITE(6,7004) CAT00810
7004 FORMAT(2X,41HPRINT SUMMARY OF SPLIT FOR EACH ITERATION) CAT00820
      IF(NAL.NE.0) WRITE(6,7009) NAL CAT00830
7009 FORMAT(2X,47HALLOCATIONS LIMITED TO VARIABLES WITH AT LEAST ,14,2X CAT00840
      1,12H OBSERVATIONS ) CAT00850
      IF(R2.GE.0.001) WRITE(6,9011) R2 CAT00860
9011 FORMAT(2X,57HALLOCATIONS LIMITED TO VARIABLES WITH A RSQ GREATER TCAT00870
      1HAN ,F4.2) CAT00880
      IF(INCON.EQ.1) WRITE(6,9012) CAT00890
9012 FORMAT(2X,26HCONSISTENCY CHECK NOT USED ) CAT00900
      IF(IDWTG.NE.0) WRITE(6,7010) IDWTG CAT00910
7010 FORMAT(2X,9H VARIABLE ,14,2X,26HUSED AS A WEIGHTING FACTOR ) CAT00920
      IF(IT.EQ.1) WRITE(6,9003) CAT00930
9003 FORMAT(2X,23HDATA INPUT IS FROM TAPE) CAT00940
      IF(NOV.LE.80.AND.IP.LE.80) GO TO 7 CAT00950
      WRITE(6,8023) CAT00960
8023 FORMAT(39HNUMBER OF VARIABLES OUTSIDE OF LIMITS. ) CAT00970
      GO TO 9999 CAT00980
7 IF(NTGC.EQ.0) GO TO 108 CAT00990
      WRITE(6,7014) CAT01000
7014 FORMAT(//2X,21HTRANSGENERATIONS USED //2X,18H K CODE I J OR C ) CAT01010
      DO 1000 I=1,NTGC CAT01020
C CAT01030
      READ TRANSGENERATION CARD CAT01040
C CAT01050
      READ(5,8002)YMAN,(KTRANS(J,I),J=1,3),TRANS(I) CAT01060
8002 FORMAT(A4,2X,13,12,13,F6.0) CAT01070
      IF(YMAN-TRNGEN) 9006,7777,9006 CAT01080
9006 WRITE(6,8026) CAT01090
8026 FORMAT(16H0TRANSGENERATION) CAT01100
      GO TO 9004 CAT01110
7777 KIP = IP CAT01120
      IF(NVA.LT.0) KIP = NOV CAT01130
      IF(KTRANS(1,I).GT.KIP.OR.KTRANS(3,I).GT.KIP) GO TO 9777 CAT01140
      IF(KTRANS(2,I).GT.24) GO TO 9775 CAT01150
      GO TO 1000 CAT01160
9777 WRITE(6,8777) I CAT01170
8777 FORMAT(///51H VARIABLE NUMBER SPECIFIED ON TRANSGENERATION CARD ,1CAT01180
      14,12H NOT CORRECT ) CAT01190
      GO TO 9999 CAT01200
9775 WRITE(6,8775) KTRANS(2,I) CAT01210
8775 FORMAT(///29H ILLEGAL TRANSGENERATION CODE ,13,10H SPECIFIED) CAT01220
      GO TO 9999 CAT01230
1000 WRITE(6,7015) (KTRANS(J,I),J = 1,3),TRANS(I) CAT01240
7015 FORMAT(3(2X,12),2X,F7.2) CAT01250
C CAT01260
      READ LABELS CARD CAT01270
C CAT01280
108 CALL RDLBL2(NLV,IP,ALREL) CAT01290
C CAT01300
      READ VARIABLE FORMAT CARDS CAT01310
C CAT01320
      IF(NVFC.LE.10) GO TO 106 CAT01330
      WRITE(6,4000) CAT01340
4000 FORMAT(1H023X71HNUMBER OF VARIABLE FORMAT CARDS INCORRECTLY SPECIFCAT01350
      XIED, ASSUMED TO BE 1.) CAT01360
      NVFC = 1 CAT01370
106 NVFC=NVFC*20 CAT01380
      READ(5,8004) (RES(I),I=1,NVFC) CAT01390
8004 FORMAT(20A4) CAT01400
      REWIND 2 CAT01410
      N3 = N + NTC CAT01420
      DO 112 K = 1,N3 CAT01430
      M(K) = 0 CAT01440
      NINCS=K CAT01450
C CAT01460
      READ DATA CAT01470
C CAT01480
      READ(5,RES) (Y(L),L=1,NOV) CAT01490
      IF(NTGC)11,11,12 CAT01500
C CAT01510
      TRANSGENERATION OF INPUT DATA CAT01520
C CAT01530
12 CALL TRANGN CAT01540
      CONTINUE CAT01550
11 WRITE(2,900) 'Y CAT01560

```

FILE. . . CL0516 FORTRAN P1

```

1129 DO 1121 JJ = 1,IP
1121 X(K,JJ) = Y(JJ)
112 CONTINUE
REWIND 2
N1 = IP
IF(N.EQ.0) GO TO 213
113 DO 360 I=1,160
CC(I) = 0
360 C(I)=0
NYAL = 0
IF(NCD.EQ.0) GO TO 36
IF(IP-66) 351,351,352
C
C READ CONTROL/DELETE CARDS
C
351 READ(5,9011) VMAN,(C(I),I=1,IP)
8011 FORMAT(A4,2X,6511)
GO TO 361
352 READ(5,8011) VMAN,(C(I),I=1,66),DUM,(C(I),I=67,IP)
361 IF(VMAN-CUNDEL) 9010,36,9010
9010 WRITE(6,8030)
8030 FORMAT(15H0CONTROL-DELETE)
GO TO 9004
36 C(INDEX)=1
DO 162 I = 1,IP
162 C(I+IP) = 1
IF(NAL.LT.1) NAL = 1
IF(IDWTG.NE.0) C(IDWTG) = 1
IQ = IP*2
37 DO 371 I = 1,IQ
DO 371 J = 1,9
371 ALI(I,J) = 0.0
C
C BEGIN SPLIT
C
DO 14 J=1,IP
IF(C(J).EQ.1) GO TO 14
D = 0
DO 122 I=1,N
DO 1221 K = 1,IP
1221 Y(K) = X(I,K)
IF(M(I).NE.0) GO TO 122
D = D + 1
XX(D,3) = 1.0
IF(IDWTG.NE.0) XX(D,3) = Y(IDWTG)
YPL0T(D) = Y(INDEX)
YPL0T(D+N) = Y(J)
XX(D,1)=Y(J)
XX(D,2)=Y(INDEX)
122 CONTINUE
N2 = D
IF(D.LT.NAL) GO TO 21
IF(D.EQ.1) GO TO 21
N3=N2 - 1
IF(NPS.EQ.1) GO TO 121
C
C PLOT RAW DATA
C
WRITE(6,1211) J,ALBEL(J)
1211 FORMAT(1H1,22HRAW DATA FOR VARIABLE ,I4,2X,A6/)
CALL SETPLT(N)
121 K=0
C
C SORT DATA INTO ASCENDING ORDER
C
DO 13 I=1,N3
A =XX(I,1)
AA=XX(I,2)
AAA = XX(I,3)
R =XX(I+1,1)
RR=XX(I+1,2)
RRR = XX(I+1,3)
IF(R.GE.A) GO TO 13
K=1
XX(I,1) = B
XX(I,2)=BB
XX(I,3) = RBB
XX(I+1,1)=A

```

CAT01570
CAT01580
CAT01590
CAT01600
CAT01610
CAT01620
CAT01630
CAT01640
CAT01650
CAT01660
CAT01670
CAT01680
CAT01690
CAT01700
CAT01710
CAT01720
CAT01730
CAT01740
CAT01750
CAT01760
CAT01770
CAT01780
CAT01790
CAT01800
CAT01810
CAT01820
CAT01830
CAT01840
CAT01850
CAT01860
CAT01870
CAT01880
CAT01890
CAT01900
CAT01910
CAT01920
CAT01930
CAT01940
CAT01950
CAT01960
CAT01970
CAT01980
CAT01990
CAT02000
CAT02010
CAT02020
CAT02030
CAT02040
CAT02050
CAT02060
CAT02070
CAT02080
CAT02090
CAT02100
CAT02110
CAT02120
CAT02130
CAT02140
CAT02150
CAT02160
CAT02170
CAT02180
CAT02190
CAT02200
CAT02210
CAT02220
CAT02230
CAT02240
CAT02250
CAT02260
CAT02270
CAT02280
CAT02290
CAT02300
CAT02310
CAT02320
CAT02330
CAT02340

FILE. . . CL0516 FORTRAN P1

```

XX(I,2)=AA
XX(I,3) = AAA
13  CONTINUE
    IF(K.EQ.1) GO TO 121
    DO 133 I= 1,N3
    NN=11
    XA = 0.0
    XB = 0.0
    XBAR = 0.0
    YBAR = 0.0
    SUMXY = 0.0
    SUMXSQ = 0.0
    SUMYSQ = 0.0
    Y1=0.0
    Y2=0.0
    Y1C = 0.0
    Y2C = 0.0
    NNN= NN+1
    XN=0
    IF (XX(NN,1).GE.XX(NNN,1)) GO TO 133
    DO 135 I= 1,NN
    XN=XN+XX(I,3)
    XA = XA + XX(I,1)*XX(I,3)
    Y1=Y1+XX(I,2)*XX(I,3)
    Y1C = Y1C + XX(I,3)
135  YBAR=YBAR+XX(I,2)*XX(I,3)
    DO 136 I=NNN,N2
    XN=XN+XX(I,3)
    XBAR=XBAR+XX(I,3)
    XB = XB + XX(I,1)*XX(I,3)
    Y2 = Y2+XX(I,2)*XX(I,3)
    Y2C = Y2C + XX(I,3)
136  YBAR=YBAR+XX(I,2)*XX(I,3)
    XBAR=XBAR/XN
    YBAR=YBAR/XN
    Y1=Y1/Y1C
    Y2=Y2/Y2C
    XA = XA/Y1C
    XB = XB/Y2C
    DO 137 I= 1,NN
    SUMXY=SUMXY + (0.0-XBAR)*(XX(I,2)-YBAR)*XX(I,3)
    SUMXSQ=SUMXSQ + ((0.0-XBAR)**2)*XX(I,3)
137  SUMYSQ=SUMYSQ + ((XX(I,2)-YBAR)**2)*XX(I,3)
    DO 138 I=NNN,N2
    SUMXY=SUMXY + (1.0-XBAR)*(XX(I,2)-YBAR)*XX(I,3)
    SUMXSQ=SUMXSQ + ((1.0-XBAR)**2)*XX(I,3)
138  SUMYSQ=SUMYSQ + ((XX(I,2)-YBAR)**2)*XX(I,3)
    R = SQRT(SUMXSQ*SUMYSQ)
    IF(R.GT.0.001) R = SUMXY/R
    RSQ=R**2
    IF(RSQ.LT.ALI(J,7)) GO TO 133
    ALI(J,7)=RSQ
    ALI(J,2)=XX(11,1)
    ALI(J,1)=1.0
7234 ALI(J,9) = (XX(11+1,1) + XX(11,1))/2.0
    ALI(J,8) = XA
    I1=3
    I2=5
    IF(Y2.GT.Y1) GO TO 1382
    ALI(J,1)=-1.0
7235 ALI(J,9) = (XX(11+1,1) + XX(11,1))/2.0
    ALI(J,8) = XB
    I1=5
    I2=3
1382 ALI(J,11)=Y1
    ALI(J,12)=Y2
    ALI(J,11+1)=NNN-1
    ALI(J,12+1)=N2-NN
133  CONTINUE
14  CONTINUE
    IF(NS.NE.1) GO TO 168
    WRITE(6,103)
103  FORMAT(1H1,4X,17HSUMMARY OF SPLITS//)
    WRITE(6,107)
107  FORMAT(//23X,8HCRITICAL,5X,6H0 MEAN,5X,6HNUMBER,9X,6HNUMBER/9X,
163H1VARIABLE SIGN LEVEL X Y CASES 1 MEAN CASES RSQ
2)
    DO 15 I=1,IP

```

CAT02350
 CAT02360
 CAT02370
 CAT02380
 CAT02390
 CAT02400
 CAT02410
 CAT02420
 CAT02430
 CAT02440
 CAT02450
 CAT02460
 CAT02470
 CAT02480
 CAT02490
 CAT02500
 CAT02510
 CAT02520
 CAT02530
 CAT02540
 CAT02550
 CAT02560
 CAT02570
 CAT02580
 CAT02590
 CAT02600
 CAT02610
 CAT02620
 CAT02630
 CAT02640
 CAT02650
 CAT02660
 CAT02670
 CAT02680
 CAT02690
 CAT02700
 CAT02710
 CAT02720
 CAT02730
 CAT02740
 CAT02750
 CAT02760
 CAT02770
 CAT02780
 CAT02790
 CAT02800
 CAT02810
 CAT02820
 CAT02830
 CAT02840
 CAT02850
 CAT02860
 CAT02870
 CAT02880
 CAT02890
 CAT02900
 CAT02910
 CAT02920
 CAT02930
 CAT02940
 CAT02950
 CAT02960
 CAT02970
 CAT02980
 CAT02990
 CAT03000
 CAT03010
 CAT03020
 CAT03030
 CAT03040
 CAT03050
 CAT03060
 CAT03070
 CAT03080
 CAT03090
 CAT03100
 CAT03110
 CAT03120

ORIGINAL PAGE IS
 OF POOR QUALITY

FILE. . . CL0516 FORTRAN P1

```

104 IF (C(I).EQ.1) GO TO 15
15 WRITE(6,104) ALBEL(I),1,(ALI(I,J),J=1,2),ALI(I,8),(ALI(I,J),J=3,7)
15 FORMAT(2X,A6,3X,13,F8.0,F8.2,F9.2,F8.2,F5.0,F9.2,F6.0,F5.2,10F2.0)
15 CONTINUE
15 BEGIN ALLOCATION
168 XMIN=99998.0
15 IVAR=-1
15 NI=IP
15 DO 17 I=1,10
15 IF (C(I).EQ.1) GO TO 17
15 IF (ALI(I,7).LT.R2) GO TO 17
15 IF (CC(I).NE.0) GO TO 170
169 IF (ALI(I,4).LT.NAL) GO TO 17
15 GO TO 171
170 IVA = ALI(I,1)
15 IF (IVA.EQ.CC(I)) GO TO 169
15 IF (ALI(I,4).LT.NAL) GO TO 17
15 IF (ALI(I,3).GT.XMIN) GO TO 17
15 IVAR = I * IP
15 GO TO 176
171 IF (ALI(I,3).GT.XMIN) GO TO 17
15 IVAR=I
176 IV = I
15 XMIN=ALI(I,3)
17 CONTINUE
15 IF (IVAR.EQ.-1) GO TO 21
15 C(IVAR) = C(IV)
15 CC(IVAR)=ALI(IV,1)
15 KVAR=IVAR
15 IF (IVAR.GT.NI) KVAR = IVAR - NI
15 DO 191 I=1,N
15 DO 1791 K=1,IP
1791 Y(K) = X(I,K)
15 IF (M(I).NE.0) GO TO 191
15 IF (ALI(KVAR,1).LT.0.0) GO TO 183
15 C VARIABLE ALLOCATED POSITIVE
15 IF (Y(KVAR) .GT.ALI(KVAR,2)) GO TO 191
179 M(I) = IVAR
15 GO TO 191
15 C VARIABLE ALLOCATED NEGATIVE
183 IF (Y(KVAR) .LE.ALI(KVAR,2)) GO TO 191
15 GO TO 179
191 CONTINUE
15 NVAL=NVAL+1
15 AL(NVAL,1)=IVAR
15 DO 20 I=1,9
15 AL(NVAL,I+1)=ALI(KVAR,I)
20 NPS = 1
15 GO TO 37
15 C INITIAL ALLOCATION COMPLETE
21 WRITE(6,109)
109 FORMAT(1H1,4X,19HINITIAL ALLOCATIONS)
15 WRITE(6,107)
15 DO 22 I=1,NVAL
15 L = AL(I,1)
15 J = L
15 IF (L.GT.NI) L = L - NI
15 WRITE(6,104) ALBEL(L),J,(AL(I,JJ),JJ=2,3),AL(I,9),(AL(I,JJ),JJ=4,8)
22 1)
15 WRITE(6,1092) NAL,R2
1092 FORMAT(///5X,25HVARIABLES WITH LESS THAN ,I3,26H CASES WERE NOT AL
15 LOCATED,5X,45HVARIABLES WITH SPLITS HAVING A RSQ LESS THAN ,F4,2,
15 120H WERE NOT ALLOCATED.)
15 DO 221 I = 1,10
15 DO 221 J = 1,9
221 ALI(I,J) = 0.0
15 DO 222 I = 1,NVAL
15 L = AL(I,1)
15 ALI(L,9) = AL(I,10)
15 ALI(L,5) = AL(NVAL,6)

```

CAT03130
 CAT03140
 CAT03150
 CAT03160
 CAT03170
 CAT03180
 CAT03190
 CAT03200
 CAT03210
 CAT03220
 CAT03230
 CAT03240
 CAT03250
 CAT03260
 CAT03270
 CAT03280
 CAT03290
 CAT03300
 CAT03310
 CAT03320
 CAT03330
 CAT03340
 CAT03350
 CAT03360
 CAT03370
 CAT03380
 CAT03390
 CAT03400
 CAT03410
 CAT03420
 CAT03430
 CAT03440
 CAT03450
 CAT03460
 CAT03470
 CAT03480
 CAT03490
 CAT03500
 CAT03510
 CAT03520
 CAT03530
 CAT03540
 CAT03550
 CAT03560
 CAT03570
 CAT03580
 CAT03590
 CAT03600
 CAT03610
 CAT03620
 CAT03630
 CAT03640
 CAT03650
 CAT03660
 CAT03670
 CAT03680
 CAT03690
 CAT03700
 CAT03710
 CAT03720
 CAT03730
 CAT03740
 CAT03750
 CAT03760
 CAT03770
 CAT03780
 CAT03790
 CAT03800
 CAT03810
 CAT03820
 CAT03830
 CAT03840
 CAT03850
 CAT03860
 CAT03870
 CAT03880
 CAT03890
 CAT03900

FILE. . . CL0516 FORTRAN P1

```

      ALI(L,1) = AL(I,2)
      ALI(L,2) = AL(I,3)
      ALI(L,3) = (ALI(L,3)*ALI(L,4) + AL(I,4)*AL(I,5))/(ALI(L,4) + AL(I,5))
      ALI(L,8) = (ALI(L,8)*ALI(L,4) + AL(I,9)*AL(I,5))/(ALI(L,4) + AL(I,5))
222  ALI(L,4) = ALI(L,4) + AL(I,5)
C
C      CONSISTENCY CHECK
C
      DO 2231 I = 1,10
      L = ALI(I,1)
2231 IF(L.EQ.0) C(I) = 1
      IF(INCON.EQ.1) GO TO 2126
      DO 2220 L = 1,NVAL
      I = AL(L,1)
      IF(C(I).EQ.2) GO TO 2120
      ALI(I,7) = (ALI(I,5) - ALI(I,3))/(ALI(I,9) - ALI(I,8))
9120 ALI(I,6) = ALI(I,3) - ALI(I,7)*ALI(I,8)
      GO TO 2220
2120 ALI(I,7) = 0.0
      ALI(I,6) = ALI(I,3)
2220 CONTINUE
      WRITE(6,2219)
2219 FORMAT(1H1,4X,17HCONSISTENCY CHECK//54X,8HORIGINAL,20X,3HNEW/5X,
186HCASE ORIGINAL NEW ACTUAL Y LIMITING X EST Y
2      DO 2222 I = 1,N
      XM1 = AL(NVAL,6)
      JJ = 0
      DO 2232 K = 1,IP
2232 Y(K) = X(I,K)
      DO 2221 J = 1,10
      IF(C(J).EQ.1) GO TO 2221
      K = J
      IF(J.GT.N1) K = J - N1
      XM = ALI(J,6) + ALI(J,7)*Y(K)
      IF(ALI(J,1).EQ.-1.AND.Y(K).LE.ALI(J,2)) XM = AL(NVAL,6)
      IF(ALI(J,1).EQ.1.AND.Y(K).GT.ALI(J,2)) XM = AL(NVAL,6)
      IF(XM.GT.XM1) GO TO 2221
      XM1 = XM
      JJ = J
2221 CONTINUE
      IF(M(I).EQ.JJ) GO TO 2222
      J = M(I)
      IF(J.EQ.0) GO TO 2222
      K = J
      IF(J.GT.N1) K = J - N1
      XM2 = ALI(J,6) + ALI(J,7)*Y(K)
      KK = JJ
      IF(JJ.GT.N1) KK = JJ - N1
      WRITE(6,2218) I,J,JJ,Y(INDEX),Y(J),XM2,Y(KK),XM1
2218 FORMAT(5X,14,4X,16,3X,16,5X,F10.5,2(5X,2F10.5))
      AT79 = ALI(J,4) - 1.0
      IF(AT79.LE..1) GO TO 2318
      ALI(J,3) = (ALI(J,3)*ALI(J,4) - Y(INDEX))/(ALI(J,4) - 1.0)
      ALI(J,8) = (ALI(J,8)*ALI(J,4) - Y(K))/(ALI(J,4) - 1.0)
      ALI(J,4) = ALI(J,4) - 1.0
      GO TO 2319
2318 ALI(J,3) = 0.
      ALI(J,8) = 0.
      ALI(J,4) = 0.
      C(J) = 1
2319 ALI(JJ,3) = (ALI(JJ,3)*ALI(JJ,4) + Y(INDEX))/(ALI(JJ,4) + 1.0)
      K = JJ
      IF(JJ.GT.N1) K = JJ - N1
      ALI(JJ,8) = (ALI(JJ,8)*ALI(JJ,4) + Y(K))/(ALI(JJ,4) + 1.0)
      ALI(JJ,4) = ALI(JJ,4) + 1.0
      M(I) = JJ
2222 CONTINUE
      WRITE(6,2225)
2225 FORMAT(1H1,4X,17HFINAL ALLOCATIONS)
      WRITE(6,107)
      DO 2226 I = 1,10
      J = I
      IF(I.GT.N1) J = I - N1
      IF(C(I).EQ.1) GO TO 2226
      WRITE(6,104) ALBEL(J,I,(ALI(I,JJ),JJ = 1,2),ALI(I,8),(ALI(I,JJ),

```

CAT03910
 CAT03920
 CAT03930
 CAT03940
 CAT03950
 CAT03960
 CAT03970
 CAT03980
 CAT03990
 CAT04000
 CAT04010
 CAT04020
 CAT04030
 CAT04040
 CAT04050
 CAT04060
 CAT04070
 CAT04080
 CAT04090
 CAT04100
 CAT04110
 CAT04120
 CAT04130
 CAT04140
 CAT04150
 CAT04160
 CAT04170
 CAT04180
 CAT04190
 CAT04200
 CAT04210
 CAT04220
 CAT04230
 CAT04240
 CAT04250
 CAT04260
 CAT04270
 CAT04280
 CAT04290
 CAT04300
 CAT04310
 CAT04320
 CAT04330
 CAT04340
 CAT04350
 CAT04360
 CAT04370
 CAT04380
 CAT04390
 CAT04400
 CAT04410
 CAT04420
 CAT04430
 CAT04440
 CAT04450
 CAT04460
 CAT04470
 CAT04480
 CAT04490
 CAT04500
 CAT04510
 CAT04520
 CAT04530
 CAT04540
 CAT04550
 CAT04560
 CAT04570
 CAT04580
 CAT04590
 CAT04600
 CAT04610
 CAT04620
 CAT04630
 CAT04640
 CAT04650
 CAT04660
 CAT04670
 CAT04680

FILE. . . CL0516 FORTRAN P1

```

1JJ=3.5)
2226 CONTINUE
2126 IF (NPA.EQ.1) GO TO 212
      DO 211 J = 1,10
      IF (C(J).EQ.1) GO TO 211
      JZ = J
      IF (J.GT.N1) JZ = J - N1
      WRITE (6,105) J,ALBEL(J),ALI(J,4)
105  FORMAT(1H1,4X,20HVARIBLE ALLOCATED ,13,2X,A6,2X,F5.0,2X,5HCASES/
      )
      WRITE (6,1051)
1051  FORMAT(/75X,4HCASE,8X,1HX,12X,1HY)
      L = 0
      L1 = ALI(J,4)
      DO 210 I = 1,N
      DO 2101 K = 1,IP
2101  Y(K) = X(I,K)
      IF (M(I).NE.J) GO TO 210
      WRITE (6,1052) I,Y(JZ),Y(INDEX)
1052  FORMAT(5X,14,2F13.4,F20.2)
      L = L + 1
      YPLOT(L) = Y(INDEX)
      YPLOT(L+N) = Y(JZ)
210  CONTINUE
      WRITE (6,105) J,ALBEL(JZ),ALI(J,4)
      CALL SETPLT(L,1)
211  CONTINUE
      GO TO 212
213  READ (5,214) NVAL,AL(NVAL,6)
214  FORMAT(13,F11.5)
      IQ = 2*IP
      DO 215 I = 1,10
      C(I) = 1
215  DO 217 I = 1,NVAL
      READ (5,216) J,ALI(J,8),ALI(J,9),ALI(J,2),ALI(J,1)
216  FORMAT(13,3F11.5,F4.0)
217  C(J) = 0
212  WRITE (6,110)
110  FORMAT(1H1,4X,13HMODEL SUMMARY/5X,55HVARIBLE NUM INTERCEPT
      )
      SLOPE CRITICAL LEVEL )
      DO 224 I = 1,10
      L = I
      IF (I.GT.N1) L=I-N1
      IF (C(I).EQ.1) GO TO 224
      IF (N.EQ.0) GO TO 2242
      IF (C(I).EQ.2) GO TO 2241
      ALI(I,9) = (ALI(I,5) - ALI(I,3))/(ALI(I,9) - ALI(I,8))
      ALI(I,8) = ALI(I,3) - ALI(I,9)*ALI(I,8)
      GO TO 2242
2241  ALI(I,9) = 0.0
      ALI(I,8) = ALI(I,3)
2242  CONTINUE
      WRITE (6,111) ALBEL(L),I,ALI(I,8),ALI(I,9),ALI(I,2)
224  CONTINUE
111  FORMAT(6X,A6,2X,13,2(2X,F11.5),6X,F11.5)
      WRITE (6,114) AL(NVAL,6)
114  FORMAT(/75X,7HMAXIMUM,F11.5)
      IF (N.EQ.0) GO TO 218
      WRITE (6,1091)
1091  FORMAT(1H1,4X,14HDEPENDENT DATA //)
      NS = 1
      CALL TEST(N,INDEX,NVAL,N1,NS)
218  IF (NTC.EQ.0) GO TO 1
      WRITE (6,1090)
1090  FORMAT(1H1,4X,9HTEST-DATA//)
      NS = N + 1
      CALL TEST(NTC,INDEX,NVAL,N1,NS)
      GO TO 1
900  FORMAT(5E15.6)
      END
      SUBROUTINE TEST(NTC,INDEX,NVAL,N1,NS)
      DIMENSION ALI(160,9),AL(160,10),KTRANS(100),KTRANS(3,100),Y(080)
      DIMENSION XMI(2),XMA(2),YPLOT(4000),C(160)
      INTEGER C
      COMMON NTC,TRANS,KTRANS,Y,N,NINCS,ALI,AL,C,XMI,XMA,YPLOT
      WRITE (6,101)
101  FORMAT(5X,60HCASE VARIABLE LIMITING X EST Y ACT Y
      )
      1 ERROR)

```

CAT04690
 CAT04700
 CAT04710
 CAT04720
 CAT04730
 CAT04740
 CAT04750
 CAT04760
 CAT04770
 CAT04780
 CAT04790
 CAT04800
 CAT04810
 CAT04820
 CAT04830
 CAT04840
 CAT04850
 CAT04860
 CAT04870
 CAT04880
 CAT04890
 CAT04900
 CAT04910
 CAT04920
 CAT04930
 CAT04940
 CAT04950
 CAT04960
 CAT04970
 CAT04980
 CAT04990
 CAT05000
 CAT05010
 CAT05020
 CAT05030
 CAT05040
 CAT05050
 CAT05060
 CAT05070
 CAT05080
 CAT05090
 CAT05100
 CAT05110
 CAT05120
 CAT05130
 CAT05140
 CAT05150
 CAT05160
 CAT05170
 CAT05180
 CAT05190
 CAT05200
 CAT05210
 CAT05220
 CAT05230
 CAT05240
 CAT05250
 CAT05260
 CAT05270
 CAT05280
 CAT05290
 CAT05300
 CAT05310
 CAT05320
 CAT05330
 CAT05340
 CAT05350
 CAT05360
 CAT05370
 CAT05380
 CAT05390
 CAT05400
 CAT05410
 CAT05420
 CAT05430
 CAT05440
 CAT05450
 CAT05460

FILE. . . CL0516 FORTRAN P1

```

      YBAR = 0.0
      REWIND 2
      IF (N5.EQ.1) GO TO 221
      N6=N5-1
220  DO 220 I=1,N6
      READ(2,900) Y
221  CONTINUE
      DO 22 I = 1,NTC
      READ(2,900) Y
      YHAR = YBAR + Y(INDEX)
      YBAR = YHAR/NTC
      TSS = 0.0
      XBAR = 0.0
      RMSE = 0.0
      REWIND 2
      IF (N5.EQ.1) GO TO 231
      DO 230 I = 1,N6
      READ(2,900) Y
230  IF (N5.EQ.1) N = NTC
231  DO 241 I=1,NTC
      READ(2,900) Y
      XMIN=AL(NVAL,6)
      JJ=INDEX
      IO = 2*N1
      DO 24 J = 1,IO
      IF (C(J).EQ.1) GO TO 24
      K = J
      IF (J.GT.N1) K = J - N1
      XM = AL(J,6) + AL(J,9)*Y(K)
      IF (AL(J,1).EQ.1.0.AND.Y(K).GT.AL(J,2)) XM = AL(NVAL,6)
      IF (AL(J,1).EQ.-1.0.AND.Y(K).LE.AL(J,2)) XM = AL(NVAL,6)
      IF (XM.GT.XMIN) GO TO 24
      XMIN = XM
      JJ=K
24  CONTINUE
      TSS = TSS + (YBAR - Y(INDEX))**2
      ERROR=Y(INDEX) -XMIN
      XBAR = XBAR + ERROR
      RMSE = RMSE + ERROR*ERROR
      Z=Y(JJ)
      YPLOT(I) = Y(INDEX)
      YPLOT(I,N) = XMIN
      IF (JJ.EQ.INDEX) Z =99999.0
      WRITE(6,1091) I,JJ,Z , XMIN,Y(INDEX) ,ERROR
1091  FORMAT(6X,13.5X,13.3(3X,F10.4),F10.4)
241  CONTINUE
      XHAR = XBAR/NTC
      ESS = TSS - RMSE
      RSQ = ESS/TSS
      XMSE = RMSE/NTC
      RMSE = SORT(XMSE)
      WRITE(6,1092) XBAR,RMSE,XMSE,RSQ
1092  FORMAT(//5X,10HMEAN ERROR,14X,F6.2/5X,22HROOT MEAN SQUARE ERROR,2X
1, F6.2/5X,17HMEAN SQUARE ERROR,7X,F6.2/
2 5X,3HRSQ,21X,F6.2)
      WRITE(6,102)
102  FORMAT(11H,5X,39HACTUAL Y ON Y-AXIS, ESTIMATE ON X-AXIS. )
      CALL SETPLT(NTC)
      RETURN
900  FORMAT(5E15.6)
      END
      SUBROUTINE POLRL2(NLBVAR,NVAR,ARRAY)
      INTEGER LABEL,ARRAY,ITEST,DUMY,BLANK
      EQUIVALENCE(LABEL,ALABEL)
      DIMENSION ARRAY(80),IDUM(7),DUMY(7)
      DATA ALABEL/3HLAB/
      DATA BLANK/4H /
      DO 1 I=1,NVAR
      APPAY(I)=BLANK
      IF (NLBVAR) 9,9,2
1  N=0
20  READ(5,3) ITEST,(IDUM(J),DUMY(J),J=1,7)
3  FORMAT(A3,3X,7(I4,A6))
      IF (ITEST-LABEL) 4,6,4
4  WRITE(6,5)
5  FORMAT(36HOLABELS CARD NOT FOUND WHEN EXPECTED)
      CALL EXIT
6  DO 8 J=1,7

```

CAT05470
 CAT05480
 CAT05490
 CAT05500
 CAT05510
 CAT05520
 CAT05530
 CAT05540
 CAT05550
 CAT05560
 CAT05570
 CAT05580
 CAT05590
 CAT05600
 CAT05610
 CAT05620
 CAT05630
 CAT05640
 CAT05650
 CAT05660
 CAT05670
 CAT05680
 CAT05690
 CAT05700
 CAT05710
 CAT05720
 CAT05730
 CAT05740
 CAT05750
 CAT05760
 CAT05770
 CAT05780
 CAT05790
 CAT05800
 CAT05810
 CAT05820
 CAT05830
 CAT05840
 CAT05850
 CAT05860
 CAT05870
 CAT05880
 CAT05890
 CAT05900
 CAT05910
 CAT05920
 CAT05930
 CAT05940
 CAT05950
 CAT05960
 CAT05970
 CAT05980
 CAT05990
 CAT06000
 CAT06010
 CAT06020
 CAT06030
 CAT06040
 CAT06050
 CAT06060
 CAT06070
 CAT06080
 CAT06090
 CAT06100
 CAT06110
 CAT06120
 CAT06130
 CAT06140
 CAT06150
 CAT06160
 CAT06170
 CAT06180
 CAT06190
 CAT06200
 CAT06210
 CAT06220
 CAT06230
 CAT06240

FILE. . . CL0516 FORTRAN P1

	K=IDUM(J)	CAT06250
	IF(K) 11,8,10	CAT06260
10	IF(K-NVAR) 7,7,11	CAT06270
11	WRITE(6,12)K,DUMY(J)	CAT06280
12	FORMAT(18H0LABELS CARD INDEX,17,18H INCORRECT. LABEL ,A6,9H IGNORE	CAT06290
	10.)	CAT06300
	GO TO 13	CAT06310
7	ARRAY(KI)=DUMY(J)	CAT06320
13	N=N+1	CAT06330
	IF(N-NLBVAR) 8,9,9	CAT06340
8	CONTINUE	CAT06350
	GO TO 20	CAT06360
9	RETURN	CAT06370
	END	CAT06380
	SUBROUTINE TRANGN	CAT06390
	DIMENSION TRANS(100),KTRANS(3,100),ALI(160,9),AL(160,10)	CAT06400
	DIMENSION X(800), C(160),XMI(2),XMA(2),YPL0T(4000)	CAT06410
	COMMON NTGC,TRANS,KTRANS,X,K,NINCS,ALI,AL,C, XMI,XMA,YPL0T	CAT06420
	INTEGER C	CAT06430
	ASN(XX)=ATAN (XX/SQRT (1.0-XX**2))	CAT06440
	NCASE = K	CAT06450
	DO 100 I=1,NTGC	CAT06460
	M=KTRANS(1,I)	CAT06470
	N=KTRANS(3,I)	CAT06480
	NTRANS=KTRANS(2,I)	CAT06490
	IF(M-81) 91,91,99	CAT06500
91	IF(N-41) 92,92,99	CAT06510
92	IF((NTRANS-25)*NTRANS)50,99,99	CAT06520
99	WRITE(6,199)I	CAT06530
199	FORMAT(22H TRANSGENERATION CARD ,I3,27H MISPUNCHED OR OUT OF ORDERCAT06540	CAT06550
	1)	CAT06560
	GO TO 100	CAT06570
50	GO TO (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23	CAT06580
	1,24),NTRANS	CAT06590
1	IF(X(N)) 198,107,108	CAT06600
107	X(M)=0.0	CAT06610
	GO TO 100	CAT06620
108	X(M)= SQRT (X(N))	CAT06630
	GO TO 100	CAT06640
2	IF(X(N)) 198,111,112	CAT06650
111	X(M)=1.0	CAT06660
	GO TO 100	CAT06670
112	X(M)= SQRT (X(N))+SQRT (X(N)+1.0)	CAT06680
	GO TO 100	CAT06690
3	IF(X(N)) 198,198,114	CAT06700
114	X(M) = ALOG(X(N))*0.4342944819	CAT06710
	GO TO 100	CAT06720
4	X(M)=EXP (X(N))	CAT06730
	GO TO 100	CAT06740
5	IF(X(N)) 198,107,117	CAT06750
117	IF(X(N)-1.0) 118,119,119	CAT06760
118	E=SQRT (X(N))	CAT06770
	X(M)=ASN(E)	CAT06780
	GO TO 100	CAT06790
119	X(M) = 3.1415926536/2.0	CAT06800
	GO TO 100	CAT06810
6	FN=NCASE	CAT06820
	E=X(M)/(FN+1.0)	CAT06830
	R=F+1.0/(FN+1.0)	CAT06840
	IF(F) 198,123,124	CAT06850
123	IF(R) 198,107,127	CAT06860
127	X(M)=ASN(SQRT (R))	CAT06870
	GO TO 100	CAT06880
124	IF(R) 198,128,129	CAT06890
128	X(M)=ASN(SQRT (E))	CAT06900
	GO TO 100	CAT06910
129	E=SQRT (E)	CAT06920
	R=SQRT (R)	CAT06930
	X(M)=ASN(E)+ASN(R)	CAT06940
	GO TO 100	CAT06950
7	IF(X(N)) 131,198,131	CAT06960
131	X(M)= 1.0/X(N)	CAT06970
	GO TO 100	CAT06980
8	X(M)=X(N)+ TRANS(I)	CAT06990
	GO TO 100	CAT07000
9	X(M)=X(N)* TRANS(I)	CAT07010
	GO TO 100	CAT07020
10	IF(X(N)) 198,107,133	

ORIGINAL PAGE IS
OF POOR QUALITY

FILE. . . CL0516 FORTRAN P1

```

133 X(M)=X(N)**TRANS(I)
GO TO 100
11 NEWB=TRANS(I)
X(M)=X(N)*X(NEWB)
GO TO 100
12 NEWB=TRANS(I)
X(M)=X(N)-X(NEWB)
GO TO 100
13 NEWB=TRANS(I)
X(M)=X(N)*X(NEWB)
GO TO 100
14 NEWB=TRANS(I)
IF(X(NEWB)) 134,197,134
134 X(M)=X(N)/X(NEWB)
GO TO 100
15 IF(X(N).GT.TRANS(I)) GO TO 111
GO TO 107
16 XNEWB=TRANS(I)
IF(X(N)-(XNEWB)) 107,111,111
17 IF(X(N)) 198,198,163
163 X(M)=ALOG(X(N))
GO TO 100
18 IF(X(N).LE.TRANS(I)) GO TO 111
GO TO 107
19 X(M)=6.11*EXP((-1.762042621E05+5.597607915E03*X(N)-2.050772636*X
1(N)**2)/(1.254162E05+273.0*X(N)))
GO TO 100
20 X(M)=SIN(X(N))
GO TO 100
21 X(M)=COS(X(N))
GO TO 100
22 IF(X(N)-1.57079632) 186,186,198
186 IF(X(N)+1.57079632) 198,187,187
187 X(M)=ATAN(X(N))
GO TO 100
23 NEWB=TRANS(I)
IF(X(N)) 198,198,188
188 X(M)=X(N)**X(NEWB)
GO TO 100
24 IF(TRANS(I)) 198,198,189
189 X(M)=TRANS(I)**X(N)
GO TO 100
197 N=NEWB
198 WRITE(
201 6.201)N,NINCS,KTRANS(2,I),M
FORMAT(23H THE VALUE OF VARIABLE ,I4, 9H IN CASE ,I5,55H VIOLATED
17HF RESTRICTIONS FOR TRANSGENERATION OF TYPE ,I3,1H./40H THE PROG
2AM CONTINUED LEAVING VARIABLE ,I4,11H UNCHANGED.)
100 CONTINUE
RETURN
END
SUBROUTINE SETPLT(N)
DIMENSION KTRANS(3,100),TRANS(100),W(080),ALI(160,9),C(150)
DIMENSION XMI(2),XMA(2),YPLT(4000),AL(160,10)
COMMON NTGC,TRANS,KTRANS,W,M,NINCS,ALI,AL,C XMI,XMA,YPLT
INTEGER C
NPV=N
NC=1
K=1
DO 330 I=1,2
XMI(I)=99999999.0
XMA(I)=-99999999.0
DO 320 J=K,NPV
XMI(I)=AMIN1(YPLT(J),XMI(I))
320 XMA(I)=AMAX1(YPLT(J),XMA(I))
K=K+M
330 NPV=NPV+M
YMAX=XMA(1)
YMIN=XMI(1)
XMAX=XMA(2)
XMIN=XMI(2)
DO 550 I=1,N
Z=YPLT(M+I)
520 Y=YPLT(I)
530 CALL PLOTR(Y,YMIN,YMAX,Z, XMIN,XMAX,NC)
550 CONTINUE
570 NC=0
580 CALL PLOTR(Y,YMIN,YMAX,Z, XMIN,XMAX,NC)

```

CAT07030
CAT07040
CAT07050
CAT07060
CAT07070
CAT07080
CAT07090
CAT07100
CAT07110
CAT07120
CAT07130
CAT07140
CAT07150
CAT07160
CAT07170
CAT07180
CAT07190
CAT07200
CAT07210
CAT07220
CAT07230
CAT07240
CAT07250
CAT07260
CAT07270
CAT07280
CAT07290
CAT07300
CAT07310
CAT07320
CAT07330
CAT07340
CAT07350
CAT07360
CAT07370
CAT07380
CAT07390
CAT07400
CAT07410
CAT07420
CAT07430
CAT07440
CAT07450
CAT07460
CAT07470
CAT07480
CAT07490
CAT07500
CAT07510
CAT07520
CAT07530
CAT07540
CAT07550
CAT07560
CAT07570
CAT07580
CAT07590
CAT07600
CAT07610
CAT07620
CAT07630
CAT07640
CAT07650
CAT07660
CAT07670
CAT07680
CAT07690
CAT07700
CAT07710
CAT07720
CAT07730
CAT07740
CAT07750
CAT07760
CAT07770
CAT07780
CAT07790
CAT07800

FILE. . . CL0516 FORTRAN P1

```

RETURN
END
SUBROUTINE PLOT(X,ZMIN,ZMAX,Y, WMIN,WMAX,NC)
DIMENSION Y(15),CLAB(12),GF(10),FMT(12),XY(51,101),SYM(15)
INTEGER XY, BLANKS
DATA SYM(1)/1H./
NP = 1
DATA TC,TP,BLANKS/1H.,1H.,1H./
DATA GF/
14H 7X.,4H 8X.,4H 9X.,4H 10X.,4H 11X.,4H 12X.,4H 13X.,4H 14X.,4H 15X.,4H 16X.,
DATA FMT/117X.,115(F1.,2.3.,'8X')/,17X.,115(F1.,2.3.,
18X),115(F12.,3),
8000 FORMAT(1H 6X5(F12.3,8X),F12.3/17X,5(F12.3,8X))
8010 FORMAT(1H F12.3,1X,103A1,F12.3)
8020 FORMAT(1H 13X,103A1)
8030 FORMAT(1H 14X,101A1)
8040 FORMAT(15X,20(5H+....),1H+)
DATA NCC/2/
IF(NCC) 10,200,10
10 KL=0
CALL SCALE(WMIN,WMAX,100.0,JY,YMIN,YMAX,YIJ)
YR=YMAX-YMIN
20 J=JY
IF(J*(J-10))70,30,30
30 IF(KL)50,50,40
40 WRITE(6,8040)
IF(KL)170,170,50
50 CLAB(1)=YMIN
DO 60 I=2,11
60 CLAB(I)=CLAB(I-1)*YIJ
WRITE(6,8000)(CLAB(I),I=1,11,2),(CLAB(J),J=2,10,2)
IF(KL)140,40,430
70 IF(J-5)90,100,80
80 J=J-5
90 JYT=5-J
100 CONTINUE
110 FMT(2)=GF(JY)
IF(KL)120,120,140
120 FMT(7)=GF(JY)
TT=JY
TT=TT*YIJ/10.0
CLAB(1)=YMIN*TT
DO 130 I=2,10
130 CLAB(I)=CLAB(I-1)*YIJ
WRITE(6,FMT)(CLAB(I),I=2,10,2),(CLAB(I),I=1,9,2)
IF(KL)140,140,430
140 IF(JY-5)150,160,150
150 WRITE(6,8030)(TC,I=1,J),(TP,(TC,I=1,4),K=1,19),TP,(TC,I=1,JYT)
IF(KL)170,170,120
160 WRITE(6,8040)
IF(KL)170,170,120
170 CONTINUE
NCC=0
IC=0
IF(INP)280,180,180
180 DO 190 I=1,51
DO 190 J=1,101
190 XY(I,J)=BLANKS
CALL SCALE(ZMIN,ZMAX,50.0,JX,XMIN,XMAX,XIJ)
XR=XMAX-XMIN
GO TO 200
ENTRY PLOTS
200 IF(INC)420,360,210
210 IF(INP)240,220,220
220 DO 270 N=1,NC
SYMB=SYM(N)
XDIFFR=XMAX-X
IF(XDIFFR)230,240,240
230 XDIFFR=0.0
240 YDIFFR=YMAX-Y(N)
IF(YDIFFR)250,260,260
250 YDIFFR=0.0
260 L=51.0-(50.0*XDIFFR)/XR*.5
K=101.0-(100.0*YDIFFR)/YR*.5
CALL FORM2(SYMB,XY(L,K))
270 CONTINUE
GO TO 440
280 DO 290 I=1,101

```

CAT07810
CAT07820
CAT07830
CAT07840
CAT07850
CAT07860
CAT07870
CAT07880
CAT07890
CAT07900
CAT07910
CAT07920
CAT07930
CAT07940
CAT07950
CAT07960
CAT07970
CAT07980
CAT07990
CAT08000
CAT08010
CAT08020
CAT08030
CAT08040
CAT08050
CAT08060
CAT08070
CAT08080
CAT08090
CAT08100
CAT08110
CAT08120
CAT08130
CAT08140
CAT08150
CAT08160
CAT08170
CAT08180
CAT08190
CAT08200
CAT08210
CAT08220
CAT08230
CAT08240
CAT08250
CAT08260
CAT08270
CAT08280
CAT08290
CAT08300
CAT08310
CAT08320
CAT08330
CAT08340
CAT08350
CAT08360
CAT08370
CAT08380
CAT08390
CAT08400
CAT08410
CAT08420
CAT08430
CAT08440
CAT08450
CAT08460
CAT08470
CAT08480
CAT08490
CAT08500
CAT08510
CAT08520
CAT08530
CAT08540
CAT08550
CAT08560
CAT08570
CAT08580

FILE. . . CL0516 FORTRAN P1

```

290 XY(1,1)=BLANKS
    L=1
    DO 320 N=1,NC
    SYMB=SYMIN
    YDIFFR=YMAX-Y(N)
    IF (YDIFFR) 300,310,310
300 YDIFFR=0.0
310 K=101.0-(100.0*YDIFFR)/YR*.5
320 CALL FORM2(SYMB,XY(L,K))
    IF (MOD(IC,5)) 340,330,340
330 W=TP
    GO TO 350
340 W=TC
350 WRITE (6,8010)X,W,(XY(1,N),N=1,101),W,X
    IC=IC+1
    GO TO 440
360 M=6-JX
    LL=50+M
    T=JX
    IF (5-JX) 370,370,380
370 T=0.0
380 RLAB=XMAX-(T*XIJ)/5.0
    W=TC
    K=52
    DO 410 L=M,LL
    K=K-1
    I=MOD(L,5)
    IF (I-1) 400,390,400
390 W=TP
    WRITE (6,8010)RLAB,W,(XY(K,N),N=1,101),W,RLAB
    RLAB=RLAB-XIJ
    W=TC
    GO TO 410
400 WRITE (6,8020)W,(XY(K,N),N=1,101),W
410 CONTINUE
420 KL=1
    GO TO 20
430 NCC=1
440 RETURN
END
SUBROUTINE SCALE(YMIN,YMAX,YINT,JY,TYMIN,TYMAX,YIJ)
C
C SUBROUTINE 'SCALE' CALCULATES THE SCALING FOR 'PLOTB'
C
    DIMENSION C(10)
    DATA C /1.0,1.5,2.0,3.0,4.0,5.0,7.5,10.0,15.0,20.0/
    DATA TEST / 0.76293945E-05/
10 YR=YMAX-YMIN
    TT=YR/YINT
    IF (TT.LE.0.0) TT = 1.0
    J = ALOG10(TT)*TEST
    E=10.0**J
    TT=TT/E
    I=0
    IF (TT-1.0*TEST) 20,30,30
20 TT=TT*10.0
    E=C/I*10.0
30 I=I+1
    IF (9-I) 40,50,50
40 E=E*10.0
    I=1
50 IF (TT-C(I)) 60,70,30
60 YIJ=C(I)*E
    GO TO 90
70 Y=YMIN/C(I)
    J=Y
    T=J
    IF (0.0001-ABS(T-Y)) 80,60,60
80 YIJ=C(I+1)*E
90 X=((YMAX+YMIN)/YIJ-YINT)/2.0+.00001
    K=X
    IF (K) 100,120,120
100 Y=K
    IF (X-Y) 110,120,110
110 K=K-1
120 TYMIN=K
    TYMIN=YIJ*TYMIN
    TYMAX=TYMIN+YINT*YIJ

```

CAT08590
CAT08600
CAT08610
CAT08620
CAT08630
CAT08640
CAT08650
CAT08660
CAT08670
CAT08680
CAT08690
CAT08700
CAT08710
CAT08720
CAT08730
CAT08740
CAT08750
CAT08760
CAT08770
CAT08780
CAT08790
CAT08800
CAT08810
CAT08820
CAT08830
CAT08840
CAT08850
CAT08860
CAT08870
CAT08880
CAT08890
CAT08900
CAT08910
CAT08920
CAT08930
CAT08940
CAT08950
CAT08960
CAT08970
CAT08980
CAT08990
CAT09000
CAT09010
CAT09020
CAT09030
CAT09040
CAT09050
CAT09060
CAT09070
CAT09080
CAT09090
CAT09100
CAT09110
CAT09120
CAT09130
CAT09140
CAT09150
CAT09160
CAT09170
CAT09180
CAT09190
CAT09200
CAT09210
CAT09220
CAT09230
CAT09240
CAT09250
CAT09260
CAT09270
CAT09280
CAT09290
CAT09300
CAT09310
CAT09320
CAT09330
CAT09340
CAT09350
CAT09360

FILE. . . CL0516 FORTRAN P1

```

      IF (YMAX-TYMAX-TEST)130,130,30
130  YIJJ=C(I)*E
      XT=((YMAX+YMIN)/YIJJ-YINT)/2.0+.00001
      KT=XT
      IF (KT) 140,150,150
140  YI=KT
      IF (XT.NE.YT) KT=KT-1
150  TYMINT=KT
      TYMINT=YIJJ*TYMINT
      TYMAXT=TYMINT+YINT*YIJJ
      IF (YMAX-TYMAXT.GT.TEST) GO TO 160
      TYMIN=TYMINT
      TYMAX=TYMAXT
      YIJ=YIJJ
      K=KT
160  TT=YINT/10.0
      JY=TT+.000001
      YIJ=YINT*(YIJ/10.0)
      J=TYMIN/YIJ
      IF (K)170,180,180
170  J=J-1
180  J=J*JY+JY-K
      JY=J
      RETURN
      END
      SUBROUTINE FORM2(SYMB,XY)
      INTEGER XY,SYMB,BLANK/' ',TEST(18)/'2','3','4','5','6','7',
1  '7','8','9','A','B','C','D','E','F','G','H','I','J',
      IF (XY.EQ.BLANK)GO TO 20
      DO 10 I=1,17
      IF (XY.NE.TEST(I))GO TO 10
      PUT IN NEXT SYMBOL OF ARRAY FOR MULTIPLE POINTS
C      XY=TEST(I+1)
      GO TO 30
10  CONTINUE
      IF (XY.EQ.TEST(18))GO TO 30
C      IF OTHER THAN CHARACTERS IN ARRAY TEST PUT IN CHARACTER 2.
      XY=TEST(1)
      GO TO 30
C      IF BLANK, PUT IN SYMBOL
20  XY=SYMB
30  RETURN
      END

```

CAT09370
 CAT09380
 CAT09390
 CAT09400
 CAT09410
 CAT09420
 CAT09430
 CAT09440
 CAT09450
 CAT09460
 CAT09470
 CAT09480
 CAT09490
 CAT09500
 CAT09510
 CAT09520
 CAT09530
 CAT09540
 CAT09550
 CAT09560
 CAT09570
 CAT09580
 CAT09590
 CAT09600
 CAT09610
 CAT09620
 CAT09630
 CAT09640
 CAT09650
 CAT09660
 CAT09670
 CAT09680
 CAT09690
 CAT09700
 CAT09710
 CAT09720
 CAT09730
 CAT09740
 CAT09750
 CAT09760
 CAT09770
 CAT09780
 CAT09790

APPENDIX B
SAMPLE INPUT DATA

APPENDIX B SAMPLE INPUT DATA

Sample input data are as follows:

FILE. . .	H2O TEST	DATA	P1
PROBLEM	H2O 20	05 00	00 05
LABFILE0001	NUMH0002	H0003	C0004
(F2.0,F4.0,2F3.0,F4.0)		01 05.1000	1 1 01 01 20000
01	5 10 4	2.5	
02	5 10 4	2.0	
03	5 10 4	2.0	
04	5 10 4	3.0	
05	5 10 4	2.5	
06	5 10 4	3.0	
07	5 10 4	2.0	
08	5 10 4	3.5	
09	5 10 4	1.0	
10	5 10 4	3.0	
11	5 10 4	3.0	
12	5 10 4	2.0	
13	5 10 4	2.5	
14	5 10 4	3.5	
15	5 10 4	2.0	
16	5 10 4	1.0	
17	5 10 4	2.0	
18	5 10 4	1.0	
19	5 10 4	2.0	
20	5 10 4	1.0	
CONDEL			
FINISH			

APPENDIX C
SAMPLE OUTPUT DATA

APPENDIX C
SAMPLE OUTPUT DATA

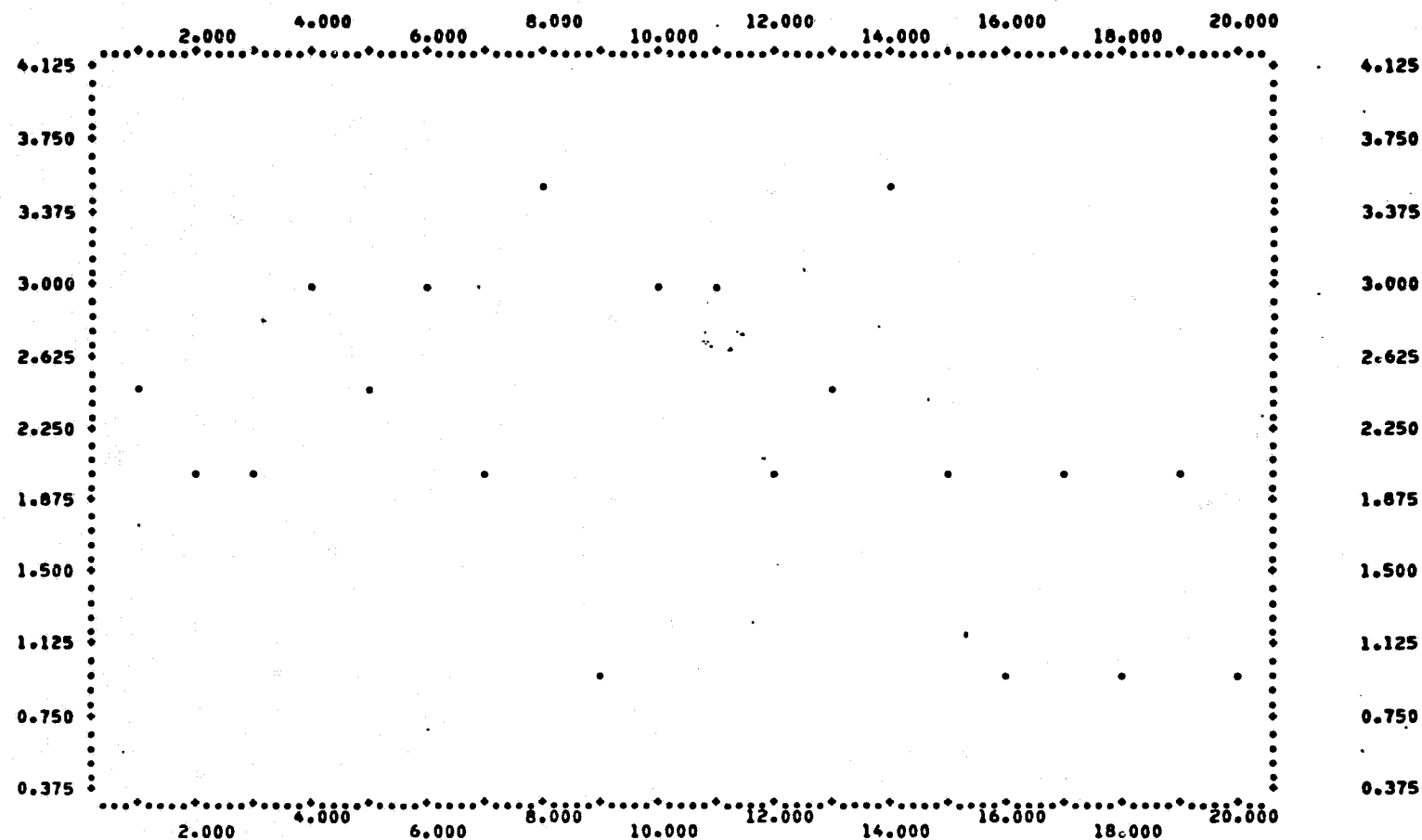
Sample output data are provided on the following pages.

CATE LIERIG CRITICAL LEVEL ANALYSIS

PROGRAM CODE	H20
NUMBER OF CASES	20
NUMBER OF ORIGINAL VARIABLES	5
INDEX OF DEPENDENT VARIABLE	5
NUMBER OF VARIABLE FORMAT CARDS	1

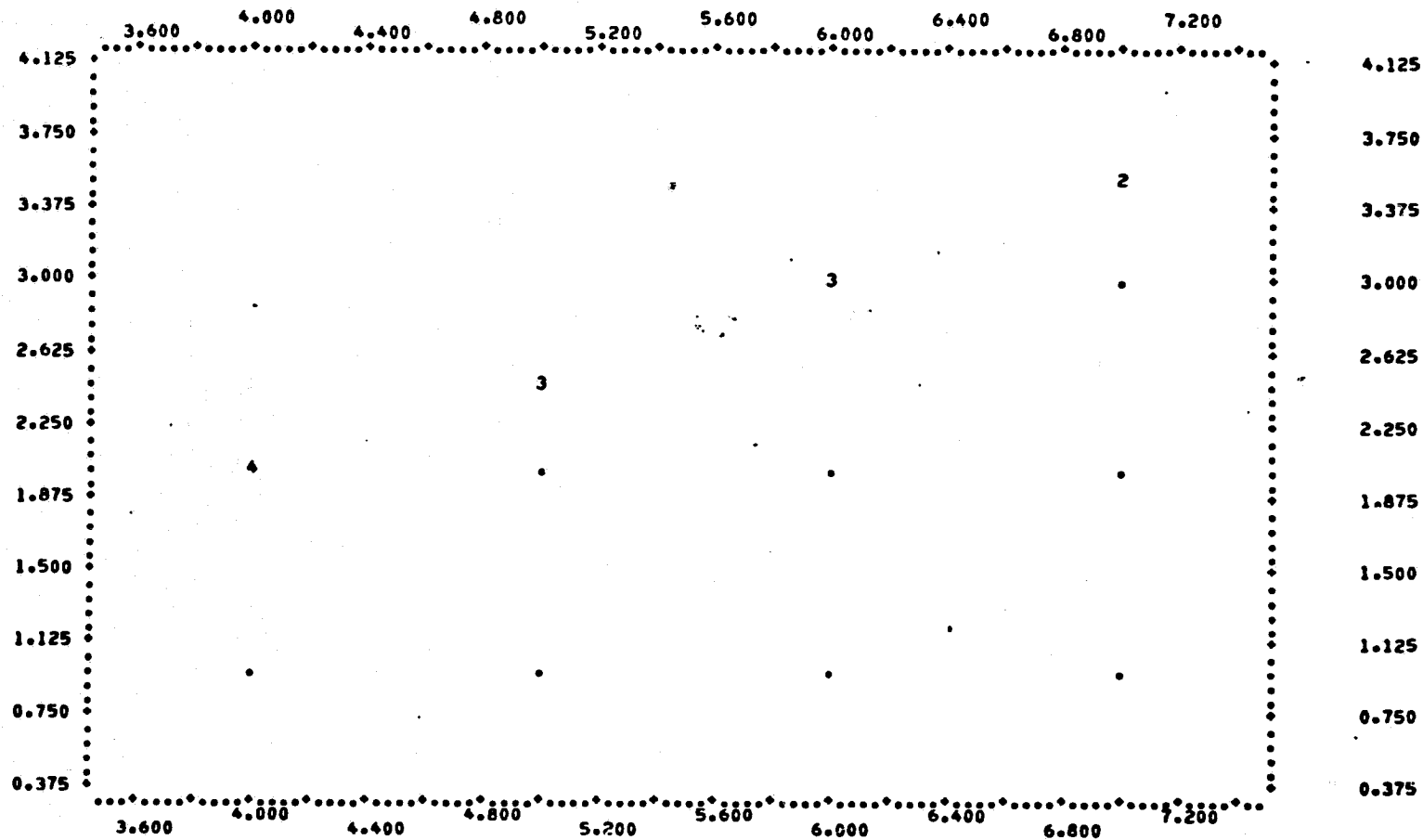
NUMBER OF VARIABLES ADDED BY TRANSGENERATION 0 TOTAL NUMBER OF VARIABLES 5
VARIABLES ARE LABELED
CONTROL DELETE CARDS USED
PLOT RAW DATA
PRINT AND PLOT EACH ALLOCATION
PRINT SUMMARY OF SPLIT FOR EACH ITERATION
ALLOCATIONS LIMITED TO VARIABLES WITH AT LEAST 2 OBSERVATIONS
ALLOCATIONS LIMITED TO VARIABLES WITH A RSD GREATER THAN 0.10

RAW DATA FOR VARIABLE 1



C-5

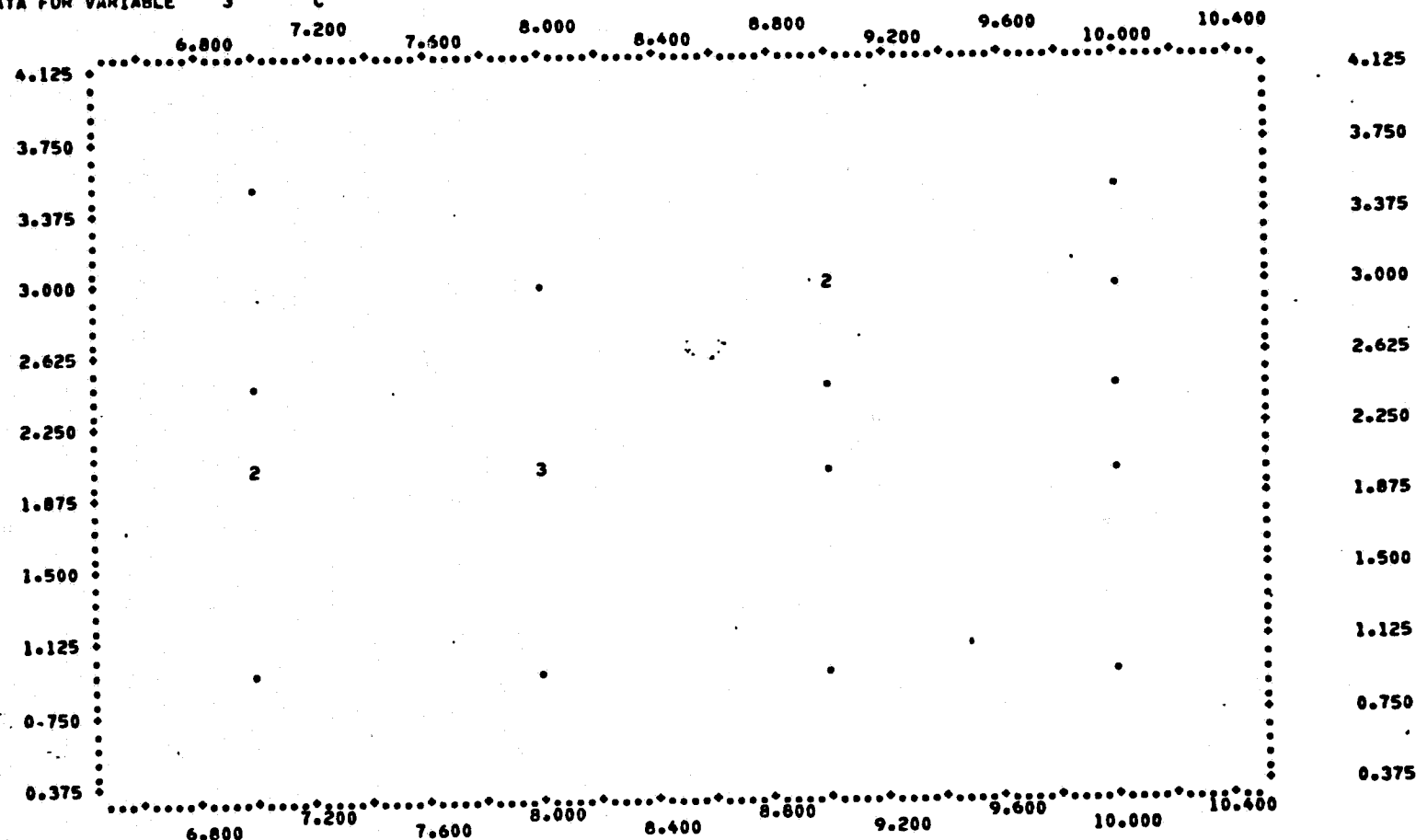
RAW DATA FOR VARIABLE 2 H



C-6

RAW DATA FOR VARIABLE 3

C

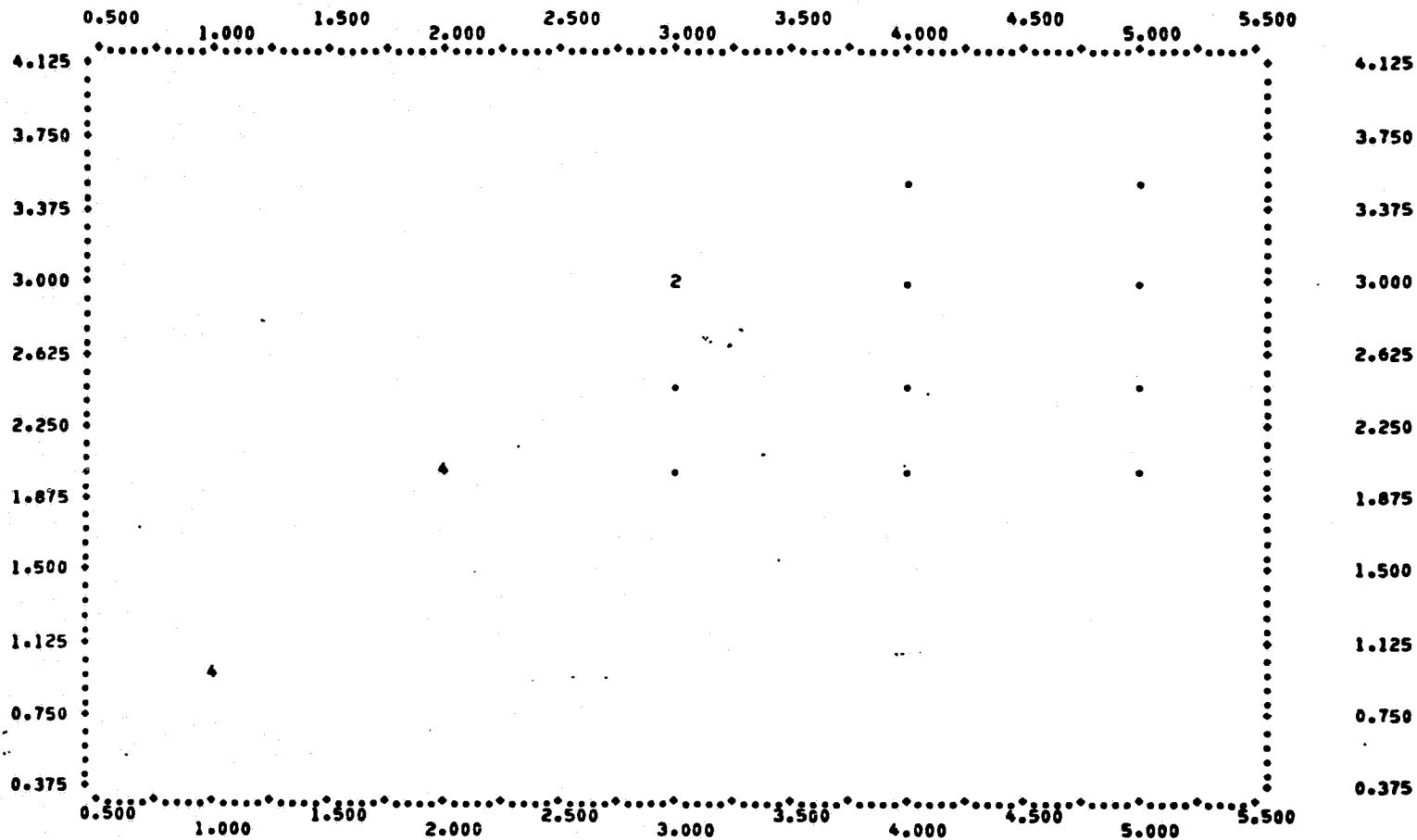


C-7

RAW DATA FOR VARIABLE

4

0



C-8

SUMMARY OF SPLITS

NUMB	VARIABLE	SIGN	CRITICAL LEVEL	0 MEAN	NUMBER CASES	1 MEAN	NUMBER CASES	RSD
H	1	-1.	15.00	18.00	1.40	15.00	15.	0.37
C	2	1.	5.00	4.50	1.95	10.00	10.	0.12
O	3	1.	8.00	7.50	2.10	10.00	10.	0.03
	4	1.	1.00	1.00	1.00	4.	16.	0.61

SUMMARY OF SPLITS

NUMB	VARIABLE	SIGN	CRITICAL LEVEL	X	0 MEAN	Y	NUMBER CASES	1	MEAN	NUMBER CASES	RSO
H	1	-1.	14.00	17.00	2.00	2.00	3.	2.65	13.	0.22	
C	2	1.	5.00	4.50	2.19	8.	2.88	8.	0.40		
O	3	1.	8.00	7.50	2.38	8.	2.69	8.	0.08		
	4	1.	2.00	2.00	2.00	4.	2.71	12.	0.32		

ORIGINAL PAGE IS
OF POOR QUALITY

SUMMARY OF SPLITS

NUMB	VARIABLE	SIGN	CRITICAL LEVEL	X	0 MEAN	Y	NUMBER CASES	1	MEAN	NUMBER CASES	RSQ
H	1	-1.	14.00	17.00	2.00	1.	2.77	11.	0.17		
C	1	1.	5.00	4.50	2.25	6.	3.17	6.	0.78		
O	1	1.	8.00	7.40	2.60	5.	2.76	7.	0.03		
	4	1.	3.00	3.00	2.63	4.	2.75	8.	0.01		

SUMMARY OF SPLITS

	VARIABLE	SIGN	CRITICAL LEVEL	0 MEAN	NUMBER CASES	1 MEAN	NUMBER CASES	RSQ
NUMB	1	1.	11.00	7.80	3.10	5.	3.50	1. 0.40
H	2	1.	6.00	6.00	3.00	3.	3.33	3. 0.50
C	3	-1.	7.00	9.20	3.10	5.	3.50	1. 0.40
O	4	1.	3.00	3.00	3.00	2.	3.25	4. 0.25

SUMMARY OF SPLITS

NUMB	VARIABLE	SIGN	CRITICAL LEVEL	0 MEAN		NUMBER CASES	1	MEAN	NUMBER CASES	RSQ
				X	Y					
H C O	1	1.	10.00	7.33	3.17	3.		3.50	1.	0.33
	2	1.	6.00	6.00	3.00	2.		3.50	2.	1.00
	3	1.	9.00	8.00	3.17	3.		3.50	1.	0.33
	4	-1.	4.00	5.00	3.25	2.		3.25	2.	0.0

SUMMARY OF SPLITS

NUMB	VARIABLE	SIGN	CRITICAL LEVEL	0 MEAN		NUMBER CASES	1 MEAN	NUMBER	
				X	Y			CASES	RSQ
BICO	4	-1.	8.00	14.00	3.50	1.	3.50	1.	0.00
		-0.	8.00	0.00	0.00	0.	0.00	0.	0.00
		-1.	7.00	10.00	3.50	1.	3.50	1.	0.00
		-1.	4.00	5.00	3.50	1.	3.50	1.	0.00

INITIAL ALLOCATIONS

	VARIABLE	SIGN	CRITICAL LEVEL	0 MEAN X	Y	NUMBER CASES	1 MEAN	NUMBER CASES	RSQ
O	4	1.	1.00	1.00	1.00	4.	2.53	16.	0.61
O	4	1.	2.00	2.00	2.00	4.	2.71	12.	0.32
O	2	1.	3.00	4.50	2.25	6.	3.17	6.	0.78
O	4	1.	3.00	3.00	3.00	2.	3.25	4.	0.25
H	2	1.	6.00	6.00	3.00	2.	3.50	2.	1.00

VARIABLES WITH LESS THAN 2 CASES WERE NOT ALLOCATED.
 VARIABLES WITH SPLITS HAVING A RSQ LESS THAN 0.10 WERE NOT ALLOCATED.

CONSISTENCY CHECK

CASE	ORIGINAL	NEW	ACTUAL Y	ORIGINAL	EST Y	NEW	EST Y
12	4	2	2.00000	2.00000	2.00000	4.00000	1.86539

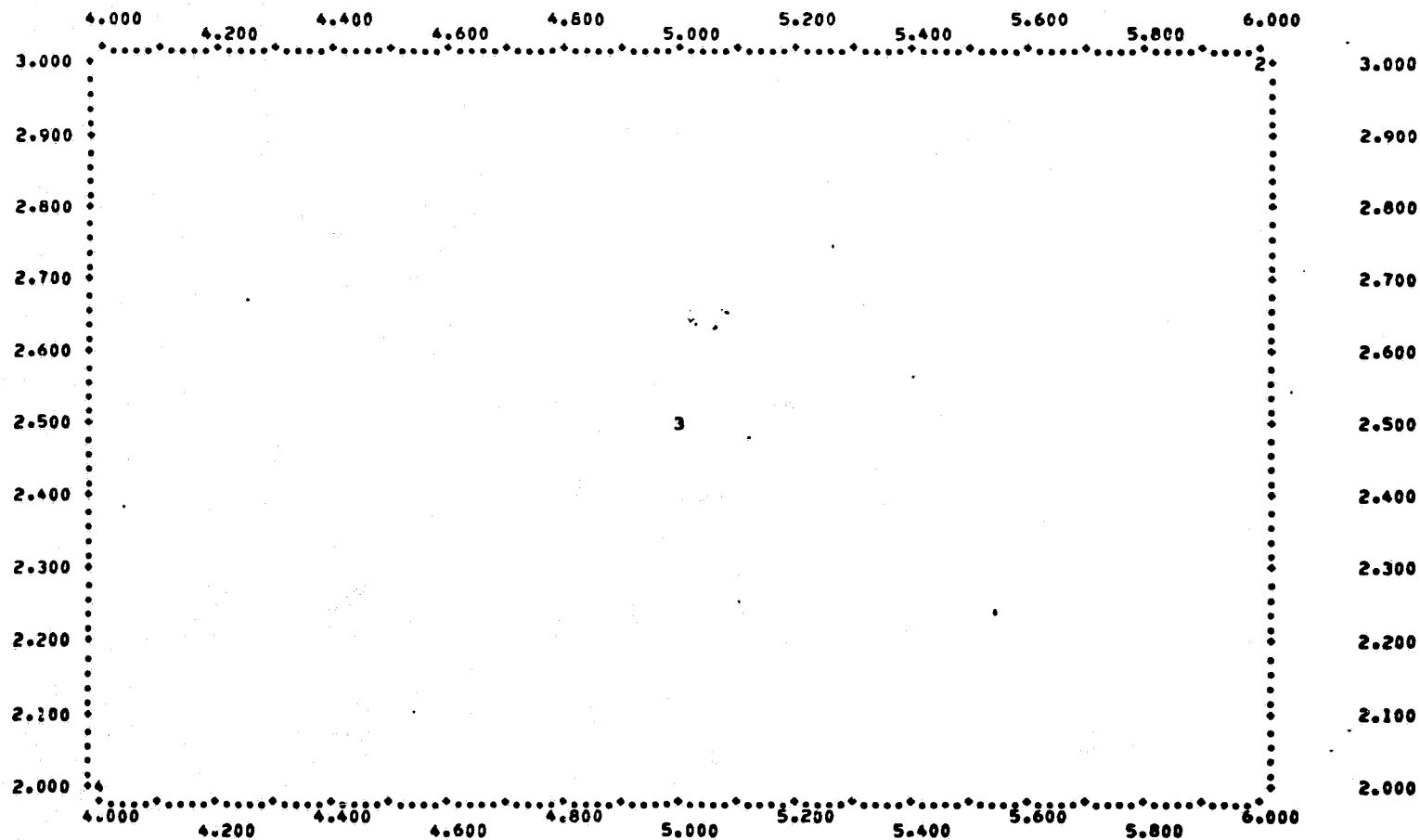
FINAL ALLOCATIONS

	VARIABLE	SIGN	CRITICAL LEVEL	X O MEAN	Y	NUMBER CASES	1 MEAN	NUMRER CASES	RSQ
H	2	1.	6.00	4.78	2.39	9.	3.50		
O	4	1.	3.00	1.78	1.78	9.	3.50		

VARIABLE ALLOCATED 2 H 9. CASES

CASE	X	Y
1	5.0000	2.5000
2	4.0000	2.0000
3	6.0000	3.0000
4	5.0000	2.5000
5	4.0000	2.0000
6	6.0000	3.0000
7	4.0000	2.0000
8	5.0000	2.5000
9	4.0000	2.0000
10	5.0000	2.5000
11	4.0000	2.0000

VARIABLE ALLOCATED 2 H 9. CASES

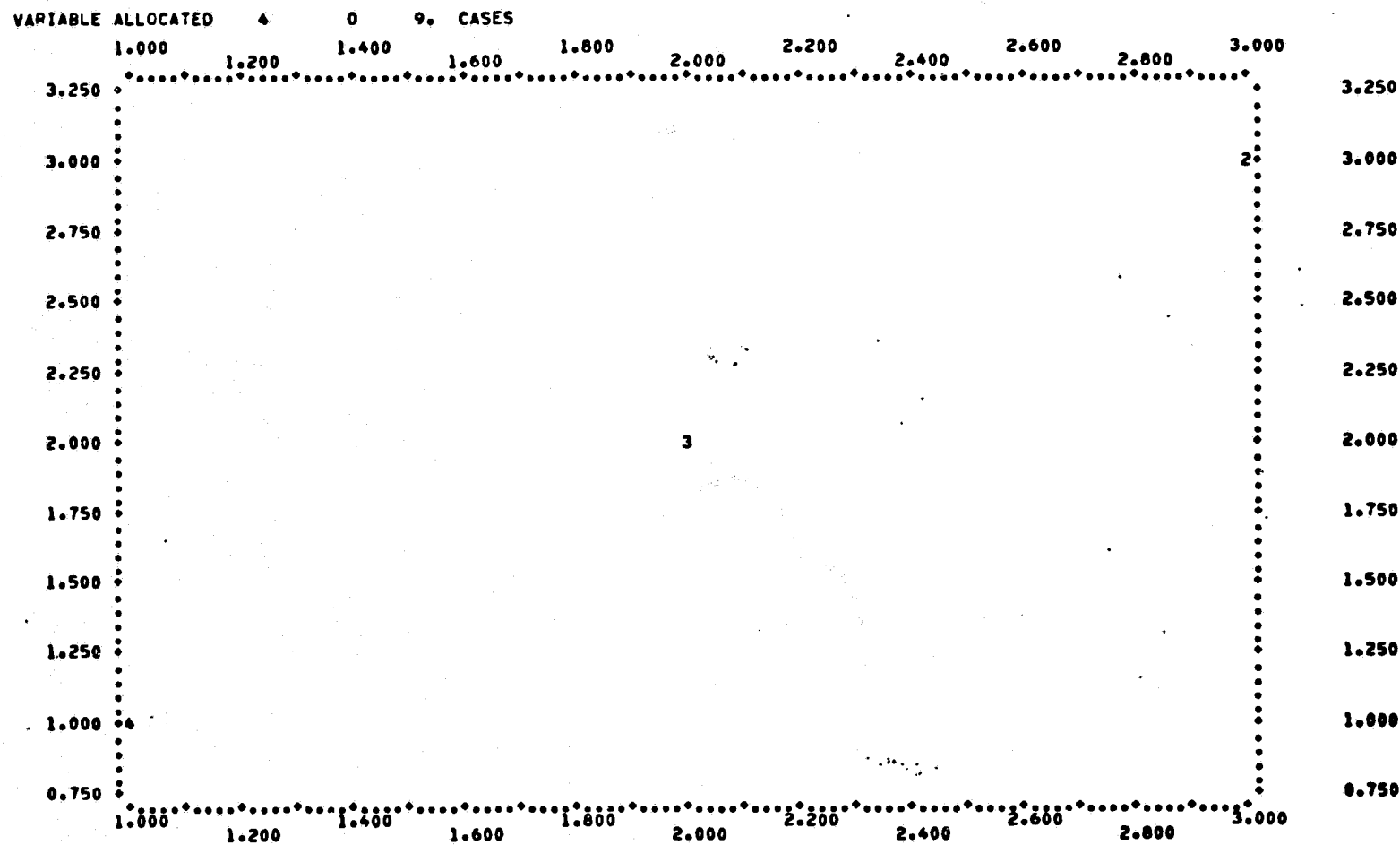


C-19

VARIABLE ALLOCATED 4 0 9. CASES

CASE	X	Y
1	2.0000	2.0000
2	3.0000	3.0000
3	1.0000	1.0000
4	2.0000	2.0000
5	2.0000	2.0000
6	1.0000	1.0000
7	2.0000	2.0000
8	1.0000	1.0000
9	2.0000	2.0000
0	1.0000	1.0000

C-21



MODEL SUMMARY		INTERCEPT	SLOPE	CRITICAL LEVEL
VARIABLE	NUM			
H	2	-0.69355	0.64516	6.00000
0	4	0.0	1.00000	3.00000
MAXIMUM		3.50000		

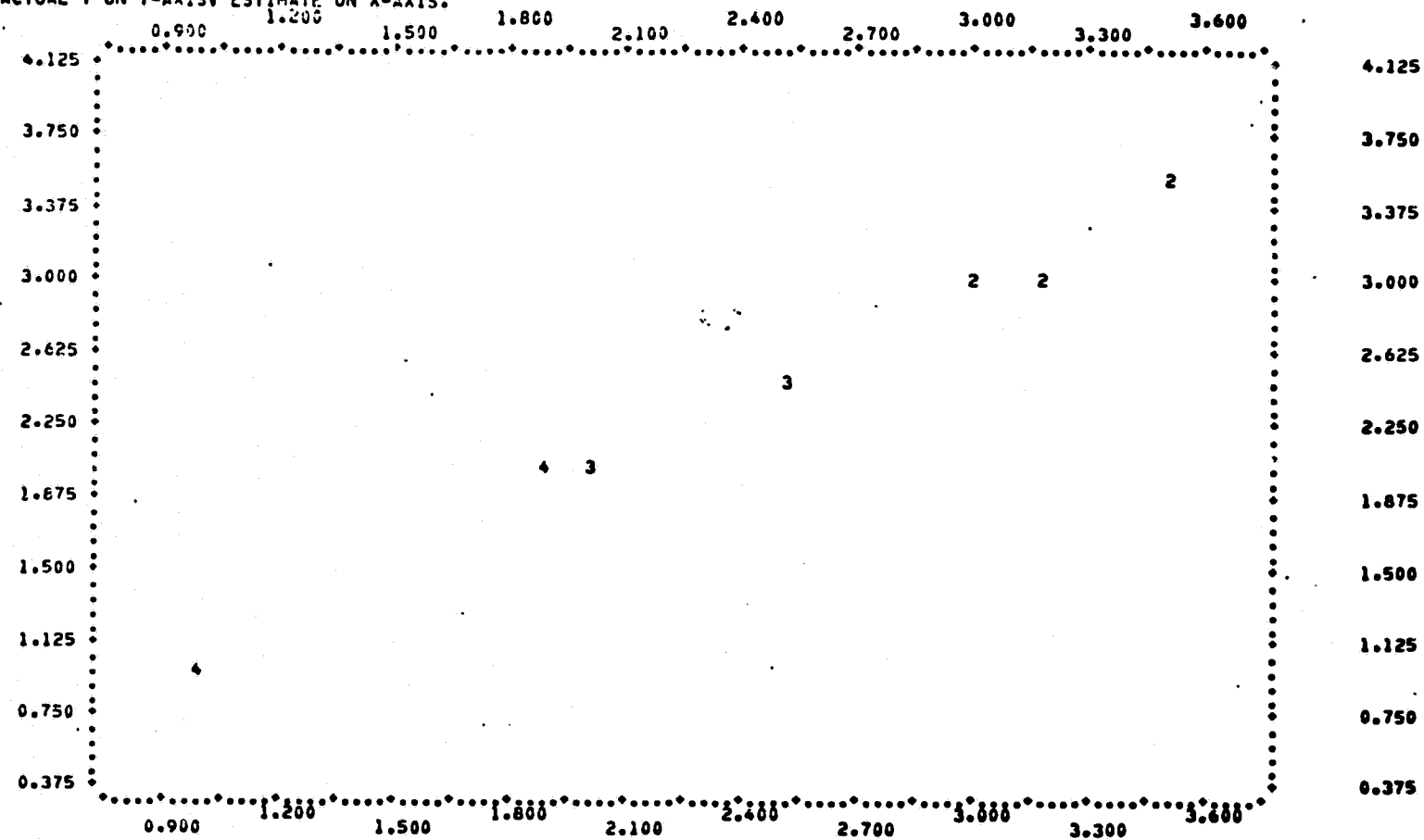
ORIGINAL PAGE IS
OF 100 & 1000000

DEPENDENT DATA

CASE	VARIABLE	LIMITING X	EST Y	ACT Y	ERROR
1	2	5.0000	2.5323	2.5000	-0.0323
2	2	4.0000	1.8871	2.0000	0.1129
3	4	2.0000	2.0000	2.0000	0.0
4	4	6.0000	3.1774	3.0000	-0.1774
5	2	5.0000	2.5323	2.5000	-0.0323
6	2	3.0000	3.0000	3.0000	0.0
7	2	4.0000	1.8871	2.0000	0.1129
8	4	4.0000	3.5000	3.5000	0.0
9	4	1.0000	1.0000	1.0000	0.0
10	2	6.0000	3.1774	3.0000	-0.1774
11	4	3.0000	3.0000	3.0000	0.0
12	2	4.0000	1.8871	2.0000	0.1129
13	2	5.0000	2.5323	2.5000	-0.0323
14	4	3.5000	3.5000	3.5000	0.0
15	4	2.0000	2.0000	2.0000	0.0
16	4	1.0000	1.0000	1.0000	0.0
17	2	4.0000	1.8871	2.0000	0.1129
18	4	1.0000	1.0000	1.0000	0.0
19	4	2.0000	2.0000	2.0000	0.0
20	4	1.0000	1.0000	1.0000	0.0

MEAN ERROR	0.00
ROOT MEAN SQUARE ERROR	0.08
MEAN SQUARE ERROR	0.01
R SQ	0.99

ACTUAL Y ON Y-AXIS, ESTIMATE ON X-AXIS.



FINISH CARD ENCOUNTERED
PROGRAM TERMINATED

C-24